

## BAB III

### METODOLOGI

#### 3.1 Alat Dan Bahan Penelitian

Pada penelitian ini membahas judul tentang “Rancang Bangun *Smart Street Lights Berbasis Internet Of Things (IoT)*” dimana perancangan dan pembuatan alat ini menggunakan metode prototipe yang meliputi perancangan perangkat keras dan perancangan perangkat lunak yang kemudian akan dilakukan pengujian serta menganalisis sistem perangkat keras dan perangkat lunak sehingga mendapatkan hasil. Pada perancangan prototipe dibutuhkan alat dan bahan yang sesuai dengan kebutuhan dan kemudian akan disusun secara teratur dan berurutan sehingga membentuk suatu alat lampu jalan pintar. Berikut tabel daftar alat dan bahan yang di butuhkan :

Tabel 3. 1 Daftar Alat dan Bahan

No	Alat dan Bahan	Type / Part Number	Jumlah
1	Laptop	Asus A455L	1
2	Solder	Dekko 20-200 Watt	1
3	Obeng	Plus 63508	1
4	Lem	Dekko DG-7	1
5	Baut	Baut TAB	5
6	Akrilik	Hitam 4 cm	5
7	Papan	±60 cm	1
8	ESP32	ESP32 Devkit V1	1
9	LED	Epistar 5 mm	5
10	Adaptor HP	VIVO 5.0 V = 1.0 A	1
11	Resistor	100 kΩ dan 230 ohm	5
12	Sensor LDR	Photoresistor	1
13	Sensor Infrared	Emitter & Detector	5
14	Kabel USB	Type B	1
15	Kendaraan Mainan	Mobil Mainan Kecil	1

#### 3.2 Alat dan Bahan

Pada penelitian ini diperlukan beberapa alat dan bahan dalam pembuatan lampu jalan pintar berbasis *Internet of Things*. Adapun alat dan bahan yang dibutuhkan adalah sebagai berikut:

**A. Alat yang digunakan dalam penelitian ini**

1. Satu set komputer/laptop berfungsi untuk pengolahan data dan pemograman
2. Solder berfungsi untuk mencairkan timah dan menyambung beberapa komponen elektronik
3. Obeng berfungsi untuk memasang lintasan
4. Lem tembak untuk merekatkan akrilik

**B. Bahan yang digunakan dalam penelitian ini**

**a) Komponen Mekanik**

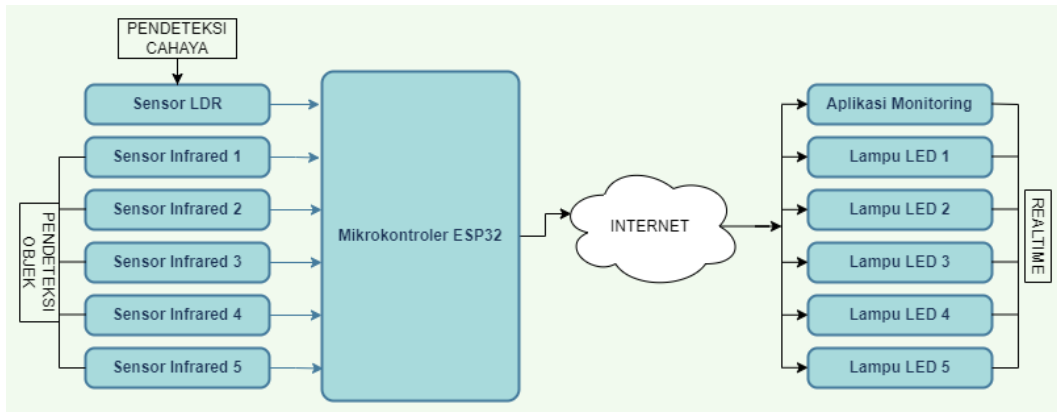
1. Mur dan Baut berfungsi sebagai penyambung komponen pada jalur lintasan
2. Akrilik berfungsi sebagai tempat sensor dan papan untuk jalan umum serta sebagai tiang lampu jalan
3. Mobil mainan berfungsi sebagai kendaraan untuk objek pengujian

**b) Komponen Elektronik**

1. ESP32 berfungsi sebagai mikrokontroler dari lampu jalan pintar
2. Lampu LED berfungsi sebagai lampu jalan
3. LM2596 berfungsi sebagai *regulator*
4. *Resistor* 100 k $\Omega$  dan *Resistor* 230  $\Omega$  berfungsi untuk pengendali arus listrik yang mengalir
5. LDR berfungsi sebagai sensor pendeteksi cahaya
6. *Infrared* berfungsi sebagai sensor pendeteksi objek
7. Kabel USB berfungsi untuk mengirimkan program ke Arduino dan juga sebagai *Port* komunikasi *serial*

**3.3 Sistem Kinerja Alat**

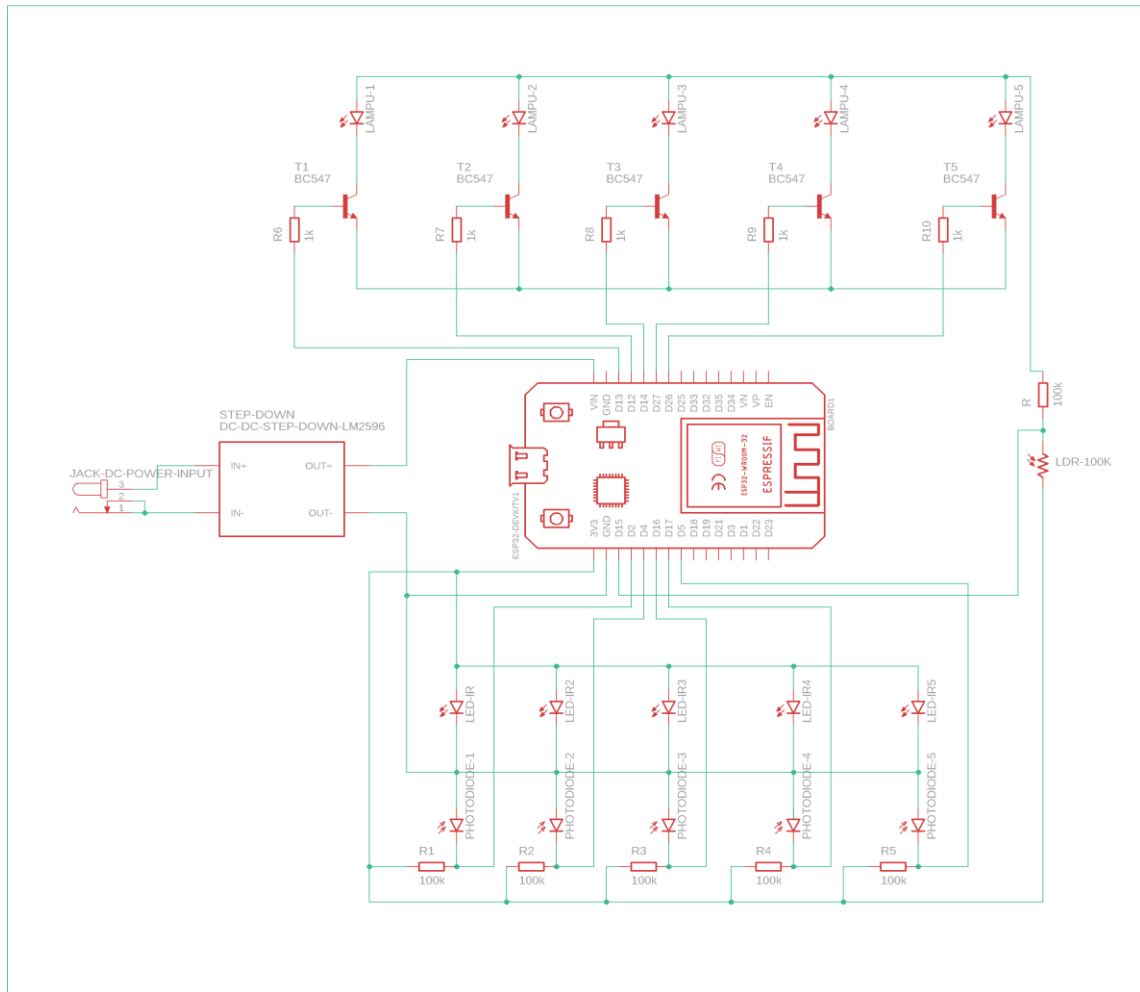
Pada rancang bangun lampu jalan pintar ini Mikrokontroler ESP32 berfungsi sebagai komponen utama atau sebagai pusat pengatur seluruh komponen lainnya seperti : Sensor LDR, Sensor *Infrared*, dan Lampu LED.



Gambar 3. 1 *Diagram Block*

Berikut penjelasan mengenai proses kinerja alat diatas, dimana cara kerja alat ini menggunakan 3 buah sensor. Sensor yang digunakan yaitu sensor LDR (mendeteksi cahaya), *infrared* (mendeteksi objek). Sensor LDR akan mendeteksi apakah kondisi saat ini sedang siang atau malam tujuannya untuk mengontrol LED agar menyala redup. Setelah itu sensor *infrared* akan mendeteksi objek untuk mentrigger agar lampu LED menyala terang. Data yang diambil sensor akan diterima oleh mikrokontroler dan akan memproses data tersebut untuk menghasilkan Lampu LED hidup dan mati secara otomatis ketika ada objek yang melewati sensor. Data yang telah diolah oleh mikrokontroler akan ditampilkan ke *database firebase* dan disinkronasikan secara *realtime*. Dalam hal ini untuk menghubungkan mikrokontroler dengan *firebase* menggunakan jaringan internet wifi. Pengujian QoS dilakukan menggunakan *platform wireshark* dengan parameter pengujian *TIPHON* yaitu *throughput*, *Packet Loss* dan *Delay*.

### 3.4 Skematik Rangkaian



Gambar 3. 2 Skematik Rangkaian

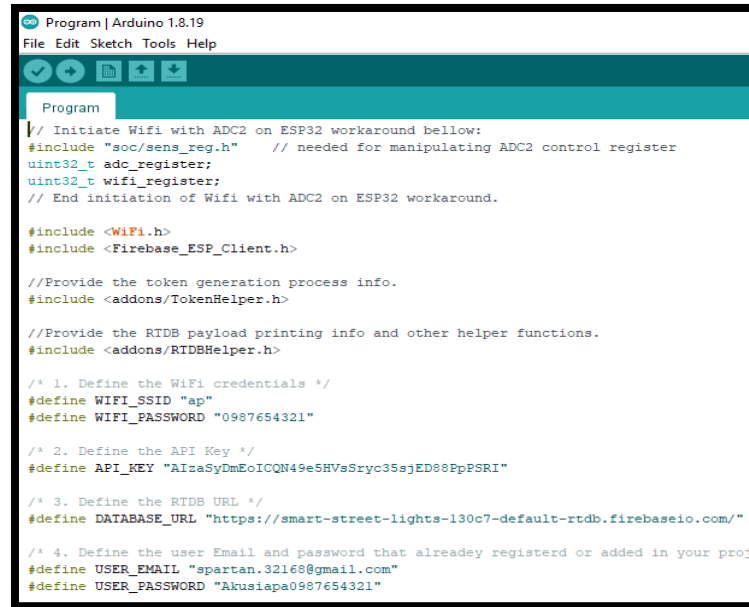
Perancangan alat pada penelitian ini menggunakan mikrokontroler untuk mengendalikan dan merespon sinyal masukan sebagai pembuat keputusan yang memberikan sinyal kepada *aktuator* sebagai tindakan. Proses komunikasi perangkat keras dapat dilakukan dengan menghubungkan pin pada masing-masing *port* mikrokontroler.

LDR digunakan sebagai sensor cahaya disambungkan kepada Pin *Analog* (A0) pada Mikrokontroler ESP32. Kaki Pin *Digital* D13, D12, D14, D27, D26 Mikrokontroler dihubungkan kepada rangkaian sistem modul LED sesuai dengan jumlah Modul LED yang digunakan yaitu sebanyak 5 buah. Sensor *Infrared* sebanyak 5 buah sesuai dengan jumlah lampu LED yang ada, berfungsi untuk mendeteksi objek yang lewat.

Untuk prinsip kerja alat ini yaitu kondisi siang hari, sensor LDR akan mendeteksi cahaya sehingga seluruh LED dalam kondisi mati. Untuk kondisi malam hari, pada saat objek (dalam hal ini mobil) melewati sensor IR 1, maka lampu LED 1 akan menyala terang. Pada saat objek melewati sensor IR 2, maka lampu LED 2 akan menyala terang dan lampu LED 1 akan kembali redup sekitar 40% dalam rentang waktu beberapa detik hal ini karena perancangan yang dilakukan dalam bentuk miniatur. Pada saat objek melewati sensor IR 3, maka lampu LED 3 akan menyala terang, lampu LED 2 akan kembali redup sekitar 40% dalam rentang waktu beberapa detik kemudian dan lampu LED 1 sudah dalam keadaan redup sekitar 40%. Pada saat objek melewati sensor IR 4, maka lampu LED 4 akan menyala terang, lampu LED 3 akan redup sekitar 40% dalam rentang waktu beberapa detik, dan lampu LED 2 akan kembali redup seperti pada lampu LED 1. Pada saat objek melewati sensor IR 5, maka lampu LED 5 akan menyala terang dan lampu LED 4 akan redup sekitar 40% dalam rentang waktu sekian detik, untuk lampu LED 3, LED 2, LED 1 dalam kondisi awal yaitu redup sekitar 40%. Program memanfaatkan PWM pada Arduino, yaitu dengan melakukan pembacaan *analog* hasil pengukuran sensor. Pertama-tama, sistem akan membaca hasil sensor LDR yang menunjukkan terjadinya perubahan tegangan akibat perubahan resistansi sensor. Terdapat sebuah batas nilai yang menunjukkan keadaan gelap. Nilai ini bergantung pada kondisi terangnya suatu lingkungan.

### 3.5 Penggunaan Software

Software pada rancang bangun sistem lampu jalan pintar berbasis *Internet of Things* menggunakan software Arduino untuk menerapkan fungsi dari mikrokontroler dan seluruh komponen rancang bangun sesuai dengan perintah yang diinput menggunakan bahasa pemrograman.



```
Program | Arduino 1.8.19
File Edit Sketch Tools Help

Program
// Initiate Wifi with ADC2 on ESP32 workaround bellow:
#include "soc/sens_reg.h" // needed for manipulating ADC2 control register
uint32_t adc_register;
uint32_t wifi_register;
// End initiation of Wifi with ADC2 on ESP32 workaround.

#include <WiFi.h>
#include <Firebase_ESP_Client.h>

//Provide the token generation process info.
#include <addons/TokenHelper.h>

//Provide the RTDB payload printing info and other helper functions.
#include <addons/RTDBHelper.h>

/* 1. Define the WiFi credentials */
#define WIFI_SSID "ap"
#define WIFI_PASSWORD "0907654321"

/* 2. Define the API Key */
#define API_KEY "AIzaSyDmEoICQN49e5HVSRyc35sjED88PpPSRI"

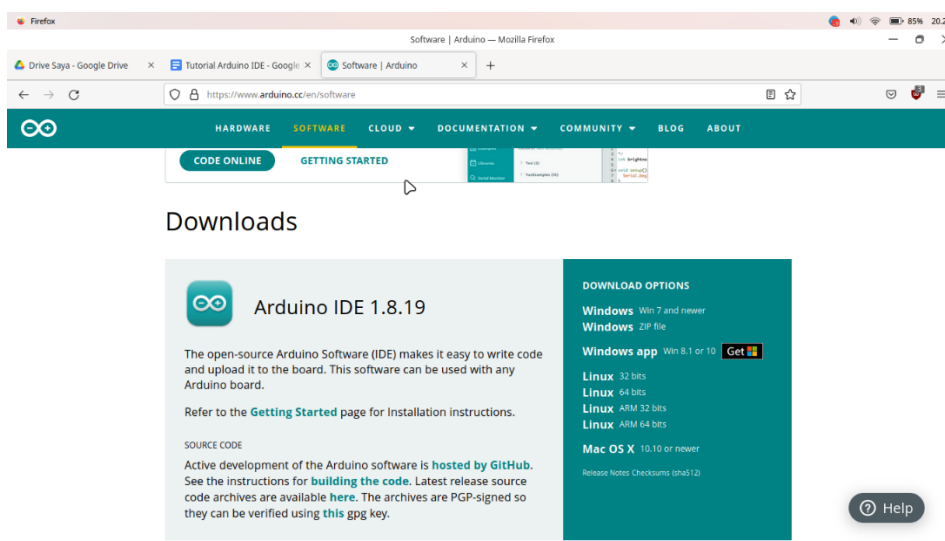
/* 3. Define the RTDB URL */
#define DATABASE_URL "https://smart-street-lights-130c7-default-rtdb.firebaseio.com/"

/* 4. Define the user Email and password that already registered or added in your project */
#define USER_EMAIL "spartan.32160@gmail.com"
#define USER_PASSWORD "Akusiapa0907654321"
```

Gambar 3. 3 Program Software Arduino

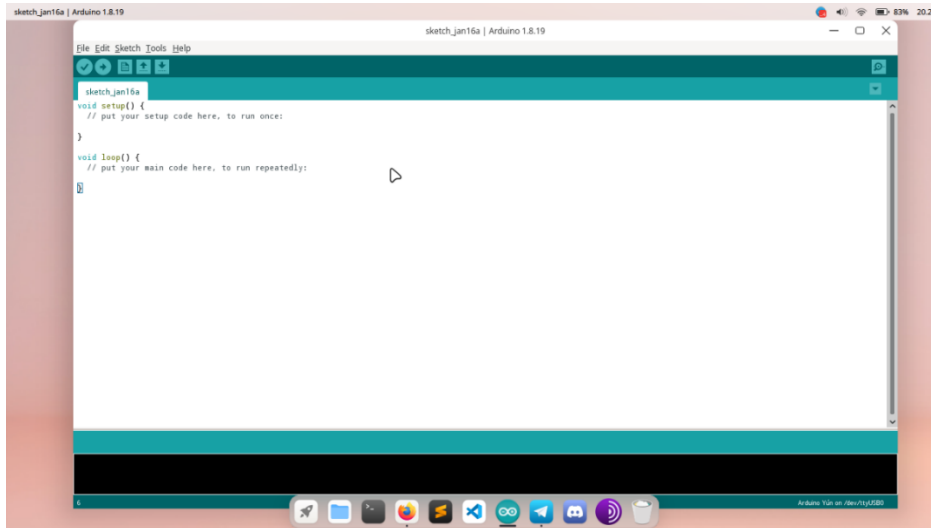
### 3.6 Langkah-langkah Arduino

1. *Download* aplikasi arduino ide <https://www.arduino.cc/en/software> sesuaikan dengan os yang digunakan.



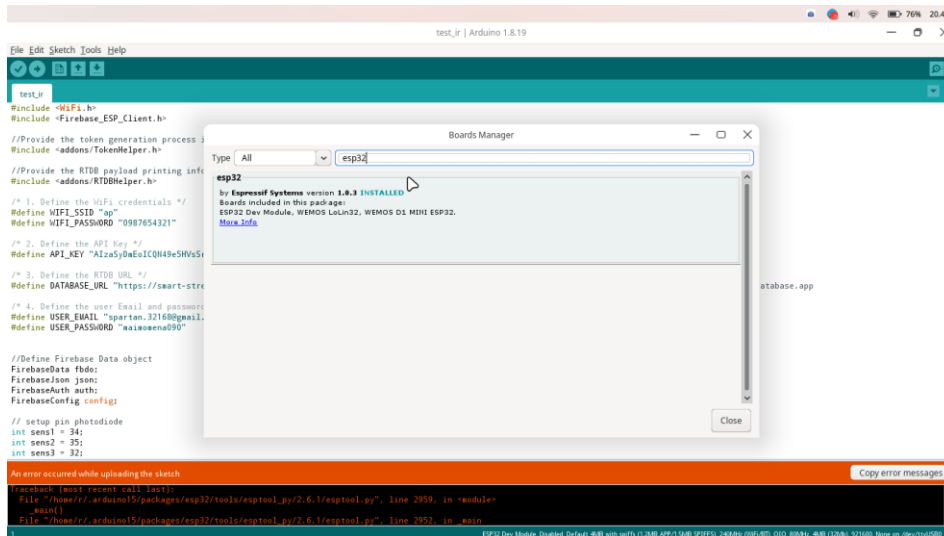
Gambar 3. 4 Aduino IDE

2. setelah aplikasi arduino ide di unduh lanjutkan dengan proses instalasinya hingga selesai.
3. setelah aplikasi terinstal, kemudian buka aplikasi tersebut, nantinya akan muncul tampilan awal seperti di bawah ini



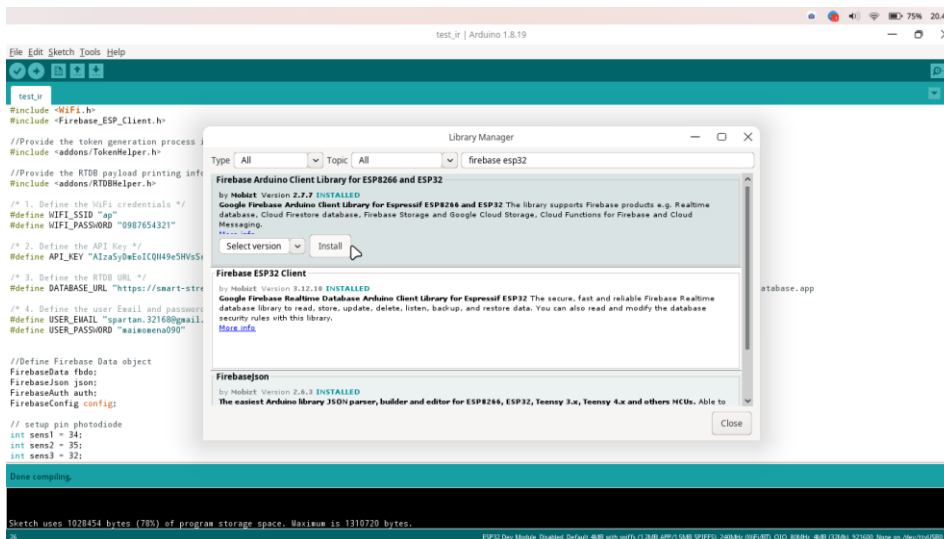
Gambar 3. 5 Tampilan Pemograman Arduino

4. selanjutnya jika sudah muncul tampilan seperti itu silahkan tekan tombol `ctrl +`, untuk membuka menu *preferences*, lalu pada menu atau bagian *additional board manager* url masukan:  
[https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json)
5. jika sudah klik oke, selanjutnya melakukan instalasi *board* dari ESP32 dengan cara klik *tools* pada tab di atas kemudian klik *board*, lalu pilih *boards manager*, setelah itu *search* ESP32, lalu pada bagian *select version* pilih versi 1.0.3



Gambar 3. 6 Board Manager ESP32

6. tunggu hingga proses selesai, jika sudah kemudian buka atau klik tab *sketch* kemudian klik *include library* > *manage library* > *search firebase ESP32*.



Gambar 3. 7 Manager Library, Firebase ESP32

7. jika sudah buka *file* kodingan yang sudah diberikan, kemudian klik tab *tools* > *Board* > *ESP32* > *ESP32 dev module*. jangan lupa untuk menghubungkan ESP32 dengan laptop menggunakan kabel data yang tersedia. setelah itu masuk tab *tools* lagi klik *port*, lalu pilih *port* ESP32 atau yang tersedia pada *port* tersebut.
8. kemudian lakukan *upload* program pada arduino ide ke ESP32, dengan cara klik tombol panah ke kanan yang tersedia di pojok kiri atas.



### 3.7 Langkah-langkah Perhitungan *Packet Loss*

Perhitungan *packets loss* dilakukan dengan menggunakan *sintaks* “*tcp.analysis.lost\_segment*” untuk mengetahui berapa jumlah *packet loss*. *Packet Loss* merupakan tanda adanya paket yang hilang pada saat pengiriman dan penerimaan data. Untuk menghitung *packet loss* dapat dilakukan menggunakan rumus berikut.

Rumus *Packet Loss* :

*Paket yang diterima = Paket yang dikirim – paket yang hilang*

*Paket yang diterima = 30086 – 8 = 30078*

Kemudian nilai yang telah ditemukan dihitung dengan menggunakan persamaan berikut :

Perhitungan persamaan *Packet Loss*:

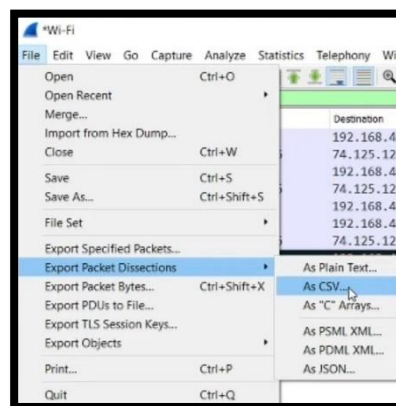
$$\text{Packet Loss} = \frac{(\text{Paket data dikirim} - \text{Paket data diterima})}{\text{Paket data yang dikirim}} \times 100\%$$

$$\text{Packet Loss} = \frac{(30086 - 30078)}{30086} \times 100\%$$

*Packet Loss = 0,0002* dibulatkan menjadi 0%

### 3.8 Langkah-langkah Perhitungan *Delay*

Perhitungan *delay* dilakukan untuk menghitung ukuran waktu yang dibutuhkan untuk pengiriman sebuah paket data dari pengirim ke penerima. Perhitungan *delay* dilakukan dengan mengekspor *file csv* yang terdapat di *wireshark* dengan cara klik menu *File -> Export Packets Dissections -> As csv ->* kemudian simpan *file* di *folder* yang diinginkan.



Gambar 3. 8 *Export Data Wireshark ke CSV*

Berikut langkah perhitungan *delay*

1. Mengubah jenis text menjadi *general* agar data bisa diolah dengan cara blok seluruh kolom A -> Menu Data text to column -> Delimeted -> Next -> Pilih Coma -> Next -> General -> Finish.

Berikut tampilan data yang telah dilakukan perubahan jenis teks.

	A	B	C	D	E	F
1	No.	Time	Source	Destination	Protocol	Length Info
2	1	0	35.201.97.85	192.168.137.104	TLSv1.2	453 Application Data
3	2	0,073077	192.168.137.104	35.201.97.85	TLSv1.2	122 Application Data
4	3	0,075228	192.168.137.104	35.201.97.85	TLSv1.2	1044 Application Data
5	4	0,077161	192.168.137.104	35.201.97.85	TLSv1.2	324 Application Data
6	5	0,078267	192.168.137.104	35.201.97.85	TLSv1.2	84 Application Data
7	6	0,079233	192.168.137.104	35.201.97.85	TLSv1.2	87 Application Data
8	7	0,080275	192.168.137.104	35.201.97.85	TLSv1.2	84 Application Data
9	8	0,297295	35.201.97.85	192.168.137.104	TCP	54 443 > 61298 [ACK] Seq=400 Ack=69 Win=65535 Len=0
10	9	0,297381	35.201.97.85	192.168.137.104	TCP	54 [TCP Dup ACK 8#1] 443 > 61298 [ACK] Seq=400 Ack=69 Win=65535 Len=0
11	10	0,297422	35.201.97.85	192.168.137.104	TCP	54 [TCP Dup ACK 8#2] 443 > 61298 [ACK] Seq=400 Ack=69 Win=65535 Len=0
12	11	0,297459	35.201.97.85	192.168.137.104	TCP	54 443 > 61298 [ACK] Seq=400 Ack=1422 Win=65535 Len=0
13	12	0,297497	35.201.97.85	192.168.137.104	TCP	54 443 > 61298 [ACK] Seq=400 Ack=69 Win=65535 Len=0
14	13	0,297537	35.201.97.85	192.168.137.104	TCP	54 [TCP Dup ACK 12#1] 443 > 61298 [ACK] Seq=400 Ack=69 Win=65535 Len=0
15	14	0,309821	35.201.97.85	192.168.137.104	TLSv1.2	453 Application Data
16	15	0,374432	192.168.137.104	35.201.97.85	TCP	54 61298 > 443 [ACK] Seq=1422 Ack=799 Win=5744 Len=0
17	16	0,382414	192.168.137.104	35.201.97.85	TLSv1.2	122 Application Data
18	17	0,384547	192.168.137.104	35.201.97.85	TLSv1.2	1044 Application Data
19	18	0,386461	192.168.137.104	35.201.97.85	TLSv1.2	324 Application Data
20	19	0,387511	192.168.137.104	35.201.97.85	TLSv1.2	84 Application Data
21	20	0,388646	192.168.137.104	35.201.97.85	TLSv1.2	87 Application Data
22	21	0,389689	192.168.137.104	35.201.97.85	TLSv1.2	84 Application Data
23	22	0,423629	35.201.97.85	192.168.137.104	TCP	54 443 > 61298 [ACK] Seq=799 Ack=1490 Win=65535 Len=0

Gambar 3. 9 Data General

2. Hapus isi kolom C sampai kolom selanjutnya.

	A	B	C
1	No.	Time	
2	1	0	
3	2	0,073077	
4	3	0,075228	
5	4	0,077161	
6	5	0,078267	
7	6	0,079233	
8	7	0,080275	
9	8	0,297295	
10	9	0,297381	
11	10	0,297422	
12	11	0,297459	
13	12	0,297497	
14	13	0,297537	
15	14	0,309821	
16	15	0,374432	
17	16	0,382414	
18	17	0,384547	
19	18	0,386461	
20	19	0,387511	
21	20	0,388646	
22	21	0,389689	
23	22	0,423629	

Gambar 3. 10 Hapus Kolom

3. Memberikan nilai *Time 1* dan *Time 2* dengan nilai yang didapat dari nilai *Time* pada kolom B.

- Membuat kolom untuk Nilai *Time* 1 dengan mengcopy nilai *time* baris pertama sampai dengan nilai *time* baris nomor 2 terakhir.
- Membuat kolom untuk Nilai *Time* 2 dengan mengcopy nilai *time* dari baris kedua sampai dengan nilai *time* terakhir.
- Membuat kolom untuk Nilai *Delay* dengan mengurangi nilai *time* 2 dengan nilai *time* 1.

	A	B	C	D	E
1	No.	Time	Time 1	Time 2	Delay
2	1	0	0	0,073077	0,073077
3	2	0,073077	0,073077	0,075228	0,002151
4	3	0,075228	0,075228	0,077161	0,001933
5	4	0,077161	0,077161	0,078267	0,001106
6	5	0,078267	0,078267	0,079233	0,000966
7	6	0,079233	0,079233	0,080275	0,001042

Gambar 3. 11 Perhitungan Nilai *Delay*

4. Menentukan jumlah nilai *delay* dengan menambahkan seluruh nilai *delay* pada kolom E dengan formula =SUM(E2:E30086)

	A	B	C	D	E
30076	30075	1048,253449	1048,253449	1048,255038	0,001589
30077	30076	1048,255038	1048,255038	1048,25588	0,000842
30078	30077	1048,25588	1048,25588	1048,256697	0,000817
30079	30078	1048,256697	1048,256697	1048,257436	0,000739
30080	30079	1048,257436	1048,257436	1048,305344	0,047908
30081	30080	1048,305344	1048,305344	1048,305433	8,9E-05
30082	30081	1048,305433	1048,305433	1048,305474	4,1E-05
30083	30082	1048,305474	1048,305474	1048,305511	3,7E-05
30084	30083	1048,305511	1048,305511	1048,305546	3,5E-05
30085	30084	1048,305546	1048,305546	1048,305583	3,7E-05
30086	30085	1048,305583	1048,305583	1051,034931	2,729348
30087	30086	1051,034931			
30088					
30089				Total Delay	1051,034931
30090					

Gambar 3. 12 Perhitungan Total *Delay*

5. Menghitung rata-rata *delay* dengan persamaan *Delay (Latency)*

Nilai untuk total paket yang diterima diketahui dengan total nilai *packet* yang terlihat pada nomor terakhir pada *file*.

$$\text{Rata - rata Delay} = \frac{\text{Total Delay}}{\text{Total Paket Yang Diterima}}$$

$$\text{Rata - rata Delay} = \frac{1051,034931}{30086} = 0,034934 \text{ s}$$

	A	B	C	D	E
30076	30075	1048,253449	1048,253449	1048,255038	0,001589
30077	30076	1048,255038	1048,255038	1048,25588	0,000842
30078	30077	1048,25588	1048,25588	1048,256697	0,000817
30079	30078	1048,256697	1048,256697	1048,257436	0,000739
30080	30079	1048,257436	1048,257436	1048,305344	0,047908
30081	30080	1048,305344	1048,305344	1048,305433	8,9E-05
30082	30081	1048,305433	1048,305433	1048,305474	4,1E-05
30083	30082	1048,305474	1048,305474	1048,305511	3,7E-05
30084	30083	1048,305511	1048,305511	1048,305546	3,5E-05
30085	30084	1048,305546	1048,305546	1048,305583	3,7E-05
30086	30085	1048,305583	1048,305583	1051,034931	2,729348
30087	30086	1051,034931			
30088					
30089				Total Delay	1051,034931
30090				Rata-Rata Delay	0,034934353

Gambar 3. 13 Perhitungan Rata-rata Delay

Untuk mengkonversikan satuan Second (s) ke millisecond (ms) dapat menggunakan rumus berikut :

$$\text{Rata-rata Delay} = \text{Nilai dari Rata-rata Delay} * 1000$$

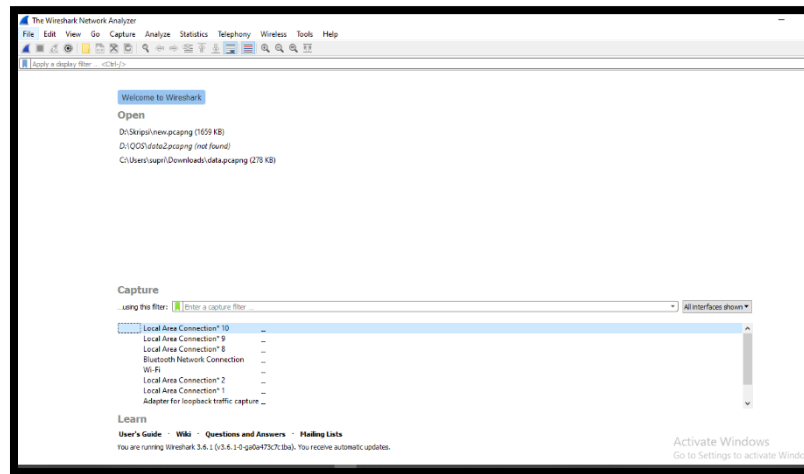
$$\text{Rata-rata Delay} = 0,034934 * 1000 = \mathbf{34,93 \text{ ms}}$$

### 3.9 Pengujian

Pengujian rancang bangun ini dilakukan dengan menggunakan seluruh komponen perangkat yang telah terintegrasi dengan bahasa pemrograman serta menggunakan objek untuk melakukan pengujian. Perangkat akan dilalui oleh sebuah objek untuk mengetahui apakah sensor, lampu LED dan seluruh komponen perangkat berhasil menerapkan fungsi sesuai perintah yang diinputkan. Adapun parameter pengujian adalah sebagai berikut:

1. Ketika sensor LDR mendeteksi adanya intensitas cahaya maka lampu jalan pintar akan otomatis mati. Sebaliknya jika sensor LDR tidak mendeteksi adanya intensitas cahaya maka lampu otomatis redup.
2. Ketika sensor *infrared* untuk mendeteksi objek maka keadaan lampu LED akan menyala terang secara bertahap.
3. Ketika sensor *infrared* tidak dilalui oleh objek maka lampu akan menyala redup.

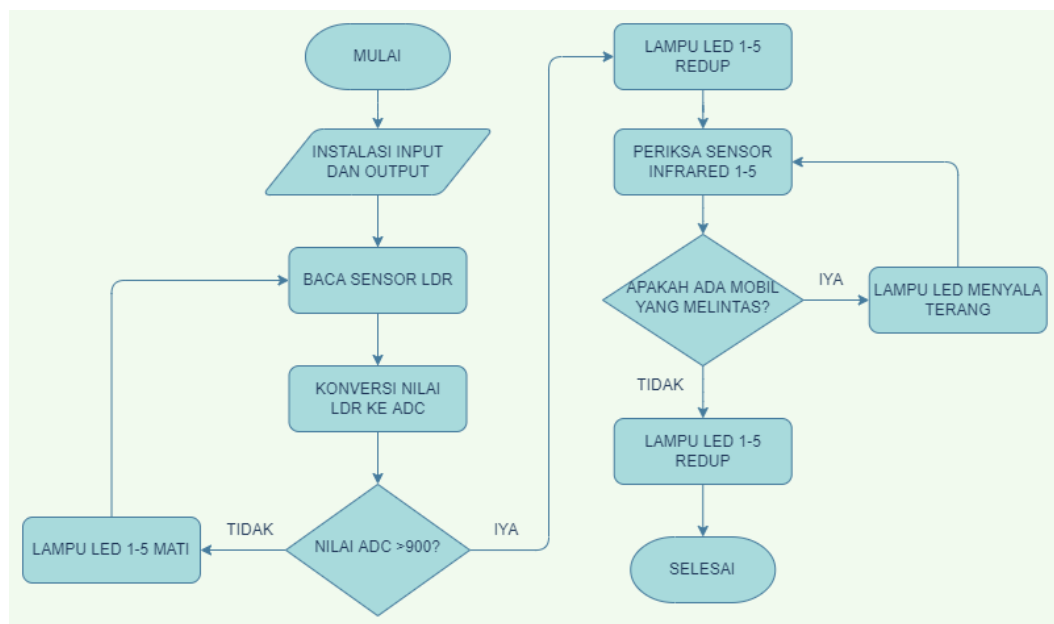
- Pengujian QoS dilakukan menggunakan *platform Wireshark* dengan parameter pengujian typhon yaitu *Throughput, Packet Loss* dan *Delay*.



Gambar 3. 14 *Software Wireshark*

### 3.10 *Flowchart* Sistem Kerja

*Flowchart* sistem merupakan bagan yang memnunjukkan alur kerja atau apa yang sedang dikerjakan didalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur- prosedur yang ada di dalam sistem. Dengan kata lain, *flowchart* ini merupakan deskripsi secara grafik dari urutan prosedur-prosedur yang terkombinasi yang membentuk suatu sistem. *Flowchart* sistem terdiri dari data yang mengalir melalui sistem dan di proses yang mentransformasikan data itu. Berikut tampilan *flowchart* sistem kerja dapat dilihat pada gambar 3.9.



Gambar 3. 15 *Flowchart* sistem kerja

### 3.9.1 Penjelasan *Flowchart* sistem kerja

Berikut penjelasan tentang *flowchart* sistem kerja. Mulai > Instalasi *input* dan *output* > Baca sensor LDR > Konversikan nilai LDR ke ADC (*Analog Digital Converter*) > Apakah nilai ADC (*Analog Digital Converter*) lebih besar dari 900? > Jika TIDAK (Sensor LDR mendeteksi adanya cahaya matahari) maka seluruh LED 1-5 akan mati > dan kembali membaca sensor LDR. Jika IYA (sensor LDR tidak mendeteksi adanya cahaya matahari) maka seluruh lampu LED 1-5 akan menyala redup > Periksa sensor *Infrared* (IR) 1-5 > Apakah ada mobil (Kendaraan) yang melintas? > Jika IYA maka lampu LED akan menyala terang dan akan kembali redup setelah 5 detik > Lalu kembali lagi membaca sensor *Infrared* (IR) 1-5 > Jika TIDAK maka lampu LED 1-5 menyala redup > Selesai.